

5.8

- (a) n 的数位长度为 $\lfloor \lg n \rfloor + 1$, 设 $k = \lfloor \lg n \rfloor + 1$.
故根据某一位数字把数分到表中去的方法做了 k 遍, 每一遍的代价为 $\Theta(n)$
故总的代价为 $\Theta(kn) = \Theta(n \lg n)$ ($k = \lfloor 2 \lg n \rfloor + 1$)
- (b) 同 (a) 分析, 时间代价为 $\Theta(2n \lg n) = \Theta(n \lg n)$
- (c) 同 (a) 分析, 时间代价为 $\Theta(n^2 \lg 2) = \Theta(n^2)$

5.9

算法 BOTTOM UPSORT 的时间复杂度为 $\Theta(n \lg n)$
此题算法 RADIX SORT 的最好时间复杂度为 $\Theta(n \lg n)$
最坏时间复杂度为: $n! = n \lg n$ 故 $\Theta(kn) = n^2 \lg n$
综上, 算法 BOTTOM UPSORT 更快.

5.11 算法 BUBBLESORT

输入: n 个元素的数组 $A[1 \dots n]$

输出: 按非降序排序的数组 $A[1 \dots n]$

sort(n)

过程 sort(i) {对 $A[i \dots n]$ 排序}

1. if $i = 1$ then

2. sort($i+1$)

3. for $j \leftarrow n$ down to $i+1$

4. if $A[j] < A[j-1]$ then

5. 交换 $A[j]$ 与 $A[j-1]$

6. end if

7. end for

8. end if

5.24

void Combination(int n , int k)

{ if ($k = 0$) Com_Output();
// Com_Output() 输出 $P[1, n]$ 中非零元素

else if ($k \leq n$)

{ $P[n] = n$;

Combinations($n-1, k-1$);

$P[n] = 0$;

Combination($n-1, k$);

}

}